

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

WHAT IS CLAIMED IS:

1. A method for using business data comprising:
 obtaining an operation to be performed on a data set and corresponding input data;
 using a SQL integration object definition to determine a structure of the data set; and
 constructing at least one SQL statement conforming to the structure to perform the operation
 on the data set according to the input data, wherein executing the at least one SQL
 statement on the data set performs the operation.

2. The method of claim 1 further comprising:
 executing each SQL statement of the at least one SQL statement on the data set once said
 each SQL statement is constructed such that said each SQL statement is executed
 prior to constructing a subsequent SQL statement of the at least one SQL statement.

3. The method of claim 1 wherein
 the SQL integration object definition comprises a plurality of SQL integration component
 definitions;
 the data set comprises a plurality of tables; and
 the SQL integration component definitions comprise:
 a parent component definition of a parent/child relationship, wherein
 the parent component definition corresponds to a parent table of the tables;
 and
 the parent component definition comprises a target key definition, the target
 key definition corresponding to a target key of the parent table;
 a child component definition of the parent/child relationship, wherein
 the child component definition corresponds to a child table of the tables; and
 the child component definition comprises a foreign key definition, the foreign
 key definition corresponding to a foreign key of the child table;
 and further comprising:
 determining that a parent record of the parent table corresponds to a parent of a child record
 of the child table when the target key of the parent record comprises a value of the
 foreign key of the child record.

4. The method of claim 1 further comprising:
 identifying an instance of the SQL integration object definition in the input data;

3 and wherein

4 the SQL integration object definition comprises a SQL integration component
5 definition;

6 the SQL integration component definition corresponds to a table of the data set;

7 the SQL integration component definition comprises a SQL integration field
8 definition;

9 the SQL integration field definition corresponds to a column of the table;

10 and

11 the constructing the at least one SQL statement comprises:

12 using the instance to identify a data value for an instance of the SQL
13 integration field definition in the input data; and

14 concatenating a clause to a first SQL statement of the at least one SQL
15 statement to select a record from the table, the record having the data
16 value for the column.

1 5. The method of claim 4 further comprising:

2 identifying a second instance of the SQL integration object definition in the input data;

3 and wherein

4 the constructing the at least one SQL statement further comprises:

5 using the second instance to identify a second data value for the SQL
6 integration field definition; and

7 concatenating an OR clause to the first SQL statement to select a second
8 record from the table, the second record having the second data value
9 for the column.

1 6. The method of claim 5 further comprising:

2 determining whether the first SQL statement comprises a maximum number of SQL clauses;

3 and

4 when the first SQL statement comprises the maximum number,

5 constructing a new SQL statement;

6 and

7 when the first SQL statement does not comprise the maximum number,

8 performing the concatenating the second clause to the first SQL statement.

1 7. The method of claim 4 wherein
 2 the identifying comprises identifying a plurality of instances; and
 3 the at least one SQL statement comprises a plurality of SQL statements;
 4 and further comprising:
 5 executing the plurality of SQL statements, wherein
 6 the executing produces a plurality of result sets;
 7 and
 8 joining the result sets to produce output data.

1 8. The method of claim 1 further comprising:
 2 executing the at least one SQL statement on the data set to produce a result set; and
 3 using the result set to produce output data.

1 9. The method of claim 1 wherein
 2 the SQL integration object definition comprises a SQL integration component definition;
 3 the SQL integration component definition corresponds to a table of the data set;
 4 the input data comprises an instance of the SQL integration object definition, the instance
 5 comprising a search specification for the SQL integration component definition;
 6 and
 7 the constructing the at least one SQL statement comprises constructing a current component
 8 SQL statement to retrieve data from the table according to the search specification.

1 10. The method of claim 1 wherein
 2 the SQL integration object definition comprises a plurality of SQL integration component
 3 definitions;
 4 the data set comprises a plurality of tables;
 5 the constructing the at least one SQL statement comprises:
 6 selecting a current component definition of the SQL integration component
 7 definitions, the current component definition corresponding to a current table
 8 of the tables; and
 9 performing the following:
 10 when the input data comprises an instance of the current component definition;
 11 generating a first SQL statement to select a record from the current
 12 table, the record corresponding to the instance;

13 when the input data does not comprise the instance and the input data
14 comprises a descendant instance of a descendant component definition
15 of the current component definition, generating a second SQL
16 statement to select all records from the current table.

1 11. The method of claim 10 further comprising:
2 executing one of the first SQL statement and the second SQL statement.

1 12. The method of claim 11 wherein
2 the executing one of the first SQL statement and the second SQL statement produces a
3 current component result set.

1 13. The method of claim 10 further comprising:
2 when the input data does not comprise the instance and the input data does not comprise the
3 descendant instance,
4 determining whether a hierarchy of output data is limited; and
5 when the hierarchy is not limited,
6 determining whether the SQL integration object definition comprises a
7 child component definition of the current component definition,
8 the child component definition corresponding to a child table of
9 the tables; and
10 when the child component definition exists,
11 generating a third SQL statement to select all records
12 from the child table.

1 14. The method of claim 13 further comprising:
2 executing one of the first SQL statement and the second SQL statement; and
3 executing the third SQL statement.

1 15. The method of claim 13 wherein
2 the determining whether the hierarchy is limited comprises determining whether a template
3 query parameter is set; and
4 when the template query parameter is set, determining that the hierarchy is limited;
5 and
6 when the template query parameter is not set, determining that the hierarchy is not

limited.

16. The method of claim 10 further comprising:
 when the input data does not comprise the instance and the input data does not comprise the descendant instance,
 determining whether a hierarchy of output data is limited; and
 when the hierarchy is not limited,
 determining whether the SQL integration object definition comprises a sibling component definition of the current component definition, the sibling component definition corresponding to a sibling table of the tables; and
 when the sibling component definition exists,
 generating a fourth SQL statement to select all records from the sibling table.

17. The method of claim 16 further comprising:
 executing one of the first SQL statement and the second SQL statement; and
 executing the fourth SQL statement.

18. The method of claim 16 wherein
 the determining whether the hierarchy is limited comprises determining whether a template query parameter is set; and
 when the template query parameter is set, determining that the hierarchy is limited;
 and
 when the template query parameter is not set, determining that the hierarchy is not limited.

19. The method of claim 1 wherein
 the SQL integration object definition comprises a plurality of SQL integration component definitions;
 the data set comprises a plurality of tables;
 the SQL integration component definitions comprise:
 a parent component definition of a parent/child relationship, the parent component definition corresponding to a parent table of the data set; and
 a child component definition of the parent/child relationship, the child component

definition corresponding to a child table of the data set;
 and
 the constructing comprises:
 constructing a current component SQL statement for the child component definition
 to select a child record of the child table corresponding to a child instance of
 the child component definition from the input data;
 using the parent/child relationship for identifying the parent component definition, the
 parent component definition determining a parent result set, the parent result
 set comprising a parent instance; and
 using the parent result set for adding a parent selection clause to the current
 component SQL statement to provide a final SQL statement.

20. The method of claim 19 further comprising:
 executing the final SQL statement.

21. The method of claim 19 wherein
 the parent component definition comprises a target key definition for a target key of the
 parent table;
 the child component definition comprises a foreign key definition for a foreign key of the
 child table; and
 the using the parent result set for adding the parent selection clause comprises:
 constructing the parent selection clause by generating a parent instance clause to
 select the child record from the child table, wherein
 the parent instance clause further selects the child record when the foreign key
 of the child record comprises a value of the target key from the parent
 instance.

22. The method of claim 1 wherein
 the constructing comprises constructing a query of the data set, wherein a result set of the
 query is used to load a SQL integration object instance as output data; the SQL
 integration object instance conforming to the SQL integration object definition.

23. The method of claim 1 wherein
 the constructing comprises constructing a SQL statement for performing one of updating,
 inserting, synchronizing and deleting data in the data set.

1 24. The method of claim 1 wherein
2 the operation comprises upserting the input data into a destination data set;
3 the SQL integration object definition comprises a SQL integration component definition;
4 the SQL integration component definition corresponds to a table of the destination data set;
5 and
6 the constructing the at least one SQL statement comprises:
7 determining whether the table comprises a record corresponding to an instance of the
8 SQL integration component definition from the input data; and
9 when the table comprises the record, constructing a first SQL statement to
10 update the record with data corresponding to the instance; and
11 when the table does not comprise the record, constructing a second SQL
12 statement to insert a new record with data corresponding to the
13 instance.

1 25. The method of claim 24 further comprising:
2 executing one of the first SQL statement and the second SQL statement.

1 26. The method of claim 1 wherein
2 the operation comprises upserting the input data into a destination data set;
3 the SQL integration object definition comprises a plurality of SQL integration component
4 definitions;
5 the data set comprises a plurality of tables;
6 a first component definition of the SQL integration component definitions comprises a parent
7 component definition of a parent/child relationship, the parent component definition
8 corresponding to a parent table of the destination data set;
9 a second component definition of the SQL integration component definitions comprises a
10 child component definition of the parent/child relationship, the child component
11 definition corresponding to a child table of the destination data set;
12 and
13 the constructing the at least one SQL statement comprises:
14 selecting a current component definition of the SQL integration component
15 definitions, the current component definition corresponding to a current table
16 of the tables; and

performing the following for the current component definition:

when the current component definition is the child component definition,

checking whether an optimized upsert of a plurality of current

component instances of the current component definition from

the input data can be performed; and

when the optimized upsert can be performed,

constructing a first set of at least one SQL statement for

performing the optimized upsert on the child

table;

when the optimized upsert cannot be performed,

constructing a second set of at least one SQL statement

to upsert the current component instances into

the child table;

when the current component definition is not the child component definition,

constructing a third set of at least one SQL statement to upsert the

current component instances into the current table.

27. The method of claim 26 further comprising:

executing one of the first set, the second set, and the third set.

28. The method of claim 25 wherein the constructing the first set comprises:

selecting a current parent instance of the parent component definition from the input data;

determining a number of child instances of the child component definition from the input

data, the child instances corresponding to children of the current parent instance;

retrieving a parent record corresponding to the current parent instance from the parent table;

determining a number of child records from the child table, wherein the child records are

children records of the parent record; and

determining a ratio of the number of children instances to the number of child records, and

when the ratio exceeds a predetermined limit,

determining in memory an associated operation of an update operation and an

insert operation for each child instance of the child instances,

and

when the ratio does not exceed the predetermined limit,

accessing the destination data set for each child instance of the child instances

15 for determining the associated operation for each child instance of the
16 child instances.

1 29. The method of claim 27 wherein
2 the determining in memory the associated operation for each child instance of the child
3 instances comprises:
4 retrieving children records of the parent record from the child table into memory; and
5 performing the following for each child instance of the child instances:
6 selecting the child instance as a current child instance of the child instances;
7 using the children records in memory to determine the associated operation for
8 the current child instance; and
9 constructing an associated SQL statement for performing the associated
10 operation on the child table.

1 30. The method of claim 29 further comprising:
2 executing the associated SQL statement for each child instance of the child instances:

1 31. The method of claim 27 wherein
2 the accessing the destination data set for each child instance of the child instances comprises
3 performing the following for each child instance of the child instances:
4 selecting the child instance as a current child instance; and
5 constructing an associated SQL statement for upserting a record corresponding
6 to the current child instance in the destination data set.

1 32. The method of claim 31 further comprising:
2 executing the associated SQL statement for each child instance of the child instances.

1 33. The method of claim 1 wherein
2 the operation is a delete record hierarchy operation;
3 the SQL integration object definition comprises a plurality of SQL integration component
4 definitions;
5 the SQL integration component definitions comprise:
6 a root component definition, the root component definition corresponding to a root
7 table of a plurality of tables of the destination data set;
8 a descendant component definition, the descendant component definition

corresponding to a descendant table of the tables, the descendant component definition corresponding to a descendant of the root component definition;
the constructing the at least one SQL statement comprises:
selecting a root instance of the root component definition from the input data;
marking a root record of the root table corresponding to the root instance;
when a cascade delete parameter is set for the descendant component definition,
selecting at least one descendant record of the descendant table, wherein
the at least one descendant record corresponds to a descendant of the
root record, and
marking the at least one descendant record;
constructing a root SQL statement for deleting the marked root record from the root table; and
constructing a descendant SQL statement for deleting the at least one marked descendant record from the descendant table.

34. The method of claim 33 further comprising:
executing the root SQL statement and the descendant SQL statement.

35. The method of claim 1 wherein
the operation comprises a reverse query operation;
the SQL integration object definition comprises a plurality of SQL integration component definitions;
the data set comprises a plurality of tables;
the SQL integration component definitions comprise:
a first parent component definition of a first parent/child relationship, the first parent component definition corresponding to a first parent table of the tables;
a first child component definition of the first parent/child relationship, the first child component definition corresponding to a first child table of tables;
and
the constructing the at least one SQL statement comprises:
constructing a first SQL statement for selecting a first child record from the first child table, wherein
the first child record corresponds to an instance of the first child component definition from the input data;

and

constructing a second SQL statement for selecting a first parent record from the first parent table, wherein the first parent record corresponds to a parent of the first child record.

36. The method of claim 35 further comprising:
executing the first SQL statement and the second SQL statement.

37. The method of claim 35 wherein
the first parent component definition comprises a target key definition, the target key definition corresponding to a target key of the first parent table;
the first child component definition comprises a foreign key definition, the foreign key definition corresponding to a foreign key of the first child table;
the first parent record corresponds to the parent of the first child record when the target key of the first parent record comprises a value of the foreign key of the first child record.

38. The method of claim 37 wherein
the SQL integration component definitions further comprise:
a third component definition, the third component definition corresponding to a third table of the tables;
the first parent component definition corresponds to a second child component definition of a second parent/child relationship,
the third component definition corresponds to a second parent component definition of the second parent/child relationship;
and further comprising:
constructing a third SQL statement for selecting a third record from the third table, the third record corresponding to a parent of the first parent record.

39. The method of claim 38 further comprising:
executing the first SQL statement, the second SQL statement, and the third SQL statement.

40. The method of claim 37 wherein
the third component definition comprises a target key definition, the target key definition corresponding to a target key of the third table;
the second child component definition comprises a foreign key definition, the foreign key

5 definition corresponding to a foreign key of the first parent table;
6 the third record corresponds to the parent of the first parent record when the target key of the
7 third record comprises a value of the foreign key of the first parent record.

1 41. The method of claim 1 wherein
2 the operation comprises a synchronize operation;
3 the data set comprises a destination data set;
4 the input data comprises a source data set; and
5 the constructing the SQL statement comprises constructing a plurality of SQL statements, the
6 constructing the plurality of SQL statements comprising:
7 constructing and executing at least one query SQL statement to query the source data
8 set for all records, wherein
9 the executing the at least one query SQL statement produces a result set;
10 constructing at least one upsert SQL statement to upsert the result set into the
11 destination data set; and
12 constructing at least one delete SQL statement for deleting an old record in the
13 destination data set that does not have a corresponding record in the source
14 data set.

1 42. The method of claim 41 further comprising:
2 executing the at least one query SQL statement, the at least one upsert SQL statement, and
3 the at least one delete SQL statement.

1 43. The method of claim 1 wherein
2 the operation comprises one of a group consisting of the following:
3 an upsert operation, wherein the corresponding input data includes an upsert instance
4 of the SQL integration object definition, wherein an update record
5 corresponding to the upsert instance is to be upserted into the data set;
6 a delete operation, wherein the corresponding input data includes a delete instance of
7 a root component definition of the SQL integration object definition, a delete
8 record corresponding to the delete instance to be deleted from the data set;
9 an update operation, wherein the corresponding input data includes an update instance
10 of the SQL integration object definition, wherein an update record
11 corresponding to the update instance is to be updated in the data set;

an insert operation, wherein the corresponding input data includes an insert instance
 of the SQL integration object definition, wherein an insert record
 corresponding to the insert instance is to be inserted into the data set;
 a query operation, wherein the corresponding input data includes a query instance of
 the SQL integration object definition, the query instance providing a first
 search specification;
 a reverse query operation, wherein the corresponding input data includes a reverse
 query instance of the SQL integration object definition, the reverse query
 instance providing a second search specification;
 a delete hierarchy operation, wherein the corresponding input data includes a root
 instance of a root component definition of the SQL integration object
 definition, wherein a root record corresponding to the root instance and
 descendants of the root record are to be deleted from the data set;
 a cascading delete operation, wherein the corresponding input data includes a root
 instance of a root component definition of the SQL integration object
 definition, wherein a root record corresponding to the root instance is to be
 deleted from the data set, and wherein a descendant record of the root record is
 to be deleted when a cascade delete parameter is set;
 a synchronize operation, wherein the corresponding input data includes a second data
 set to be synchronized with the data set; and
 an execute operation comprising at least one of the upsert operation, the update
 operation, the insert operation, the query operation, the reverse query
 operation, the delete operation, the cascading delete operation, and the
 synchronize operation.

44. A method for using business data comprising:
 obtaining an operation to be performed on a data set and corresponding input data;
 constructing at least one SQL statement to perform the operation on the data set according to
 the input data; and
 executing each SQL statement of the at least one SQL statement on the data set once said
 each SQL statement is constructed such that said each SQL statement is executed
 prior to constructing a subsequent SQL statement of the at least one SQL statement.

45. The method of claim 44 wherein

the constructing the subsequent SQL statement uses a result set of executing a prior SQL statement.

46. The method of claim 44 wherein the operation comprises a query operation.

47. The method of claim 44 wherein the operation comprises a reverse query operation.

48. The method of claim 44 wherein the operation comprises a delete record hierarchy operation.

49. The method of claim 44 wherein the operation comprises an upsert operation.

50. The method of claim 44 wherein the operation comprises a synchronize operation.

51. A computer system comprising:
a processor; and
a memory, the memory comprising instructions, the processor for executing the instructions, the instructions comprising:
obtaining instructions to obtain an operation to be performed on a data set and corresponding input data;
structure determining instructions to use a SQL integration object definition to determine a structure of the data set; and
constructing instructions to construct at least one SQL statement conforming to the structure to perform the operation on the data set according to the input data, wherein executing the at least one SQL statement on the data set performs the operation.

52. The computer system of claim 51 wherein the instructions further comprise: executing instructions to execute the at least one SQL statement.

53. The computer system of claim 51 wherein the SQL integration object definition comprises a plurality of SQL integration component definitions;

4 the data set comprises a plurality of tables; and
5 the SQL integration component definitions comprise:
6 a parent component definition of a parent/child relationship, wherein
7 the parent component definition corresponds to a parent table of the tables;
8 and
9 the parent component definition comprises a target key definition, the target
10 key definition corresponding to a target key of the parent table; and
11 a child component definition of the parent/child relationship, wherein
12 the child component definition corresponds to a child table of the tables; and
13 the child component definition comprises a foreign key definition, the foreign
14 key definition corresponding to a foreign key of the child table;
15 and further comprising:
16 parent determining instructions to determine that a parent record of the parent table
17 corresponds to a parent of a child record of the child table when the target key of the
18 parent record comprises a value of the foreign key of the child record.

1 54. The computer system of claim 51 further comprising:
2 instance identifying instructions to identify an instance of the SQL integration object
3 definition in the input data;
4 and wherein
5 the SQL integration object definition comprises a SQL integration component
6 definition;
7 the SQL integration component definition corresponds to a table of the data set;
8 the SQL integration component definition comprises a SQL integration field
9 definition;
10 the SQL integration field definition corresponds to a column of the table;
11 and
12 the constructing instructions comprise:
13 value determining instructions to use the instance to identify a data value for
14 an input instance of the SQL integration field definition in the input
15 data; and
16 concatenating instructions to concatenate a clause to a first SQL statement of
17 the at least one SQL statement to select a record from the table, the
18 record having the data value for the column.

1 55. The computer system of claim 54 wherein
2 the instance identifying instructions further identify a second input instance of the SQL
3 integration object definition in the input data; and
4 the constructing instructions further perform the following:
5 use the second instance to identify a second data value for the SQL integration field
6 definition; and
7 concatenate an OR clause to the first SQL statement to select a second record from
8 the table, the second record having the second data value for the column.

1 56. The computer system of claim 55 wherein the instructions further comprise:
2 length determining instructions to determine whether the first SQL statement comprises a
3 maximum number of SQL clauses, and
4 when the first SQL statement comprises the maximum number,
5 construct a new SQL statement,
6 and
7 when the first SQL statement does not comprise the maximum number,
8 perform the concatenating the second clause to the first SQL statement.

1 57. The computer system of claim 51 wherein
2 the SQL integration object definition comprises a plurality of SQL integration component
3 definitions;
4 the data set comprises a plurality of tables;
5 the constructing instructions comprise:
6 selecting instructions to select a current component definition of the SQL integration
7 component definitions, the current component definition corresponding to a
8 current table of the tables; and
9 generating instructions to perform the following:
10 when the input data comprises an instance of the current component definition,
11 generate a first SQL statement to select a record from the current table,
12 the record corresponding to the instance; and
13 when the input data does not comprise the instance and the input data
14 comprises a descendant instance of a descendant component definition

of the current component definition, generate a second SQL statement to select all records from the current table.

58. The computer system of claim 57 wherein the generating instructions further comprise:

hierarchy determining instructions to determine whether a hierarchy of output data is limited when the input data does not comprise the instance and the input data does not comprise the descendant instance; and

child generating instructions to perform the following:

when the hierarchy is not limited,

determine whether the SQL integration object definition comprises a

child component definition of the current component definition,

the child component definition corresponding to a child table of

the tables, and

when the child component definition exists,

generate a third SQL statement to select all records

from the child table.

59. The computer system of claim 51 wherein the SQL integration object definition comprises a plurality of SQL integration component definitions;

the data set comprises a plurality of tables;

the SQL integration component definitions comprise:

a parent component definition of a parent/child relationship, the parent component

definition corresponding to a parent table of the data set; and

a child component definition of the parent/child relationship, the child component

definition corresponding to a child table of the data set;

and

the constructing instructions further comprise:

current component constructing instructions to construct a current component SQL

statement for the child component definition to select a child record of the

child table corresponding to a child instance of the child component definition

from the input data;

parent identifying instructions to use the parent/child relationship for identifying the

parent component definition, the parent component definition determining a parent result set, the parent result set comprising a parent instance; and parent selection clause adding instructions to use the parent result set to add a parent selection clause to the current component SQL statement to provide a final SQL statement.

60. The computer system of claim 59 wherein the parent selection clause adding instructions comprise: parent instance clause generating instructions to generate a parent instance clause to select the child record from the child table, wherein the parent instance clause further selects the child record when the foreign key of the child record comprises a value of the target key from the parent instance.

61. The computer system of claim 51 wherein the operation comprises upserting the input data into a destination data set; the SQL integration object definition comprises a plurality of SQL integration component definitions; the data set comprises a plurality of tables; a first component definition of the SQL integration component definitions comprises a parent component definition of a parent/child relationship, the parent component definition corresponding to a parent table of the destination data set; a second component definition of the SQL integration component definitions comprises a child component definition of the parent/child relationship, the child component definition corresponding to a child table of the destination data set; and the constructing instructions comprise: selecting instructions to select a current component definition of the SQL integration component definitions, the current component definition corresponding to a current table of the tables; and performing instructions to perform the following for the current component definition: when the current component definition is the child component definition, check whether an optimized upsert of a plurality of current component instances of the current component definition from the input

data can be performed; and
 when the optimized upsert can be performed,
 construct a first set of at least one SQL statement for
 performing the optimized upsert on the child
 table; and
 when the optimized upsert cannot be performed,
 construct a second set of at least one SQL statement to
 upsert the current component instances into the
 child table;
 and
 when the current component definition is not the child component definition,
 construct a third set of at least one SQL statement to upsert the current
 component instances into the current table.

62. The computer system of claim 61 wherein the performing instructions to
 construct the first set comprise:
 calculating instructions to perform the following:
 select a current parent instance of the parent component definition from the input
 data;
 determine a number of child instances of the child component definition from the
 input data, the child instances corresponding to children of the current parent
 instance;
 retrieve a parent record corresponding to the current parent instance from the parent
 table;
 determine a number of child records from the child table, wherein the child records
 are children records of the parent record; and
 determine a ratio of the number of children instances to the number of child records,
 and
 when the ratio exceeds a predetermined limit,
 determine in memory an associated operation of an update operation and an
 insert operation for each child instance of the child instances,
 and
 when the ratio does not exceed the predetermined limit,
 access the destination data set for each child instance of the child instances to

determine the associated operation for each child instance of the child instances.

63. The computer system of claim 62 wherein the calculating instructions to determine in memory the associated operation for each child instance of the child instances comprise:

- retrieving instructions to retrieve children records of the parent record from the child table into memory; and
- in-memory instructions to perform the following for each child instance of the child instances:
 - select the child instance as a current child instance of the child instances;
 - use the children records in memory to determine the associated operation for the current child instance; and
 - construct an associated SQL statement for performing the associated operation on the child table.

64. The computer system of claim 51 wherein the operation is a delete record hierarchy operation; the SQL integration object definition comprises a plurality of SQL integration component definitions; the SQL integration component definitions comprise:

- a root component definition, the root component definition corresponding to a root table of a plurality of tables of the destination data set;
- a descendant component definition, the descendant component definition corresponding to a descendant table of the tables, the descendant component definition corresponding to a descendant of the root component definition;

and

the constructing instructions further comprise:

- root selecting instructions to select a root instance of the root component definition from the input data;
- root marking instructions to mark a root record of the root table corresponding to the root instance;
- descendant selecting instructions to perform the following:
 - when a cascade delete parameter is set for the descendant component

definition,
 select at least one descendant record of the descendant table wherein
 the at least one descendant record corresponds to a descendant
 of the root record, and
 mark the at least one descendant record;
 root constructing instructions to construct a root SQL statement to delete the marked
 root record from the root table; and
 descendant constructing instructions to construct a descendant SQL statement to
 delete the at least one marked descendant record from the descendant table.

65. The computer system of claim 51 wherein
 the operation comprises a reverse query operation;
 the SQL integration object definition comprises a plurality of SQL integration component
 definitions;
 the data set comprises a plurality of tables;
 the SQL integration component definitions comprise:
 a first parent component definition of a first parent/child relationship, the first parent
 component definition corresponding to a first parent table of the tables;
 a first child component definition of the first parent/child relationship, the first child
 component definition corresponding to a first child table of tables;
 and
 the constructing instructions comprise:
 first constructing instructions to construct a first SQL statement for selecting a first
 child record from the first child table, wherein
 the first child record corresponds to an instance of the first child component
 definition from the input data;
 and
 second constructing instructions to construct a second SQL statement for selecting a
 first parent record from the first parent table, wherein the first parent record
 corresponds to a parent of the first child record.

66. The computer system of claim 65 wherein
 the first parent component definition comprises a target key definition, the target key
 definition corresponding to a target key of the first parent table;

the first child component definition comprises a foreign key definition, the foreign key definition corresponding to a foreign key of the first child table; and
the first parent record corresponds to the parent of the first child record when the target key of the first parent record comprises a value of the foreign key of the first child record.

67. The computer system of claim 65 wherein
the SQL integration component definitions further comprise:
a third component definition, the third component definition corresponding to a third table of the tables;
the first parent component definition corresponds to a second child component definition of a second parent/child relationship;
the third component definition corresponds to a second parent component definition of the second parent/child relationship; and
the constructing instructions further comprise:
third constructing instructions to construct a third SQL statement for selecting a third record from the third table, the third record corresponding to a parent of the first parent record.

68. A computer system comprising:
a processor; and
a memory, the memory comprising instructions, the processor for executing the instructions, the instructions comprising:
obtaining instructions to obtain an operation to be performed on a data set and corresponding input data;
constructing instructions to construct at least one SQL statement to perform the operation on the data set according to the input data; and
executing instructions to execute each SQL statement of the at least one SQL statement on the data set once said each SQL statement is constructed such that said each SQL statement is executed prior to constructing a subsequent SQL statement of the at least one SQL statement.

69. The computer system of claim 68 wherein
the constructing instructions comprise subsequent constructing instructions to construct the subsequent SQL statement using a result set of executing a prior SQL statement.

1 70. A computer program product comprising:
 2 computer instructions comprising:
 3 obtaining instructions to obtain an operation to be performed on a data set and
 4 corresponding input data;
 5 structure determining instructions to use a SQL integration object definition to
 6 determine a structure of the data set; and
 7 constructing instructions to construct at least one SQL statement conforming to the
 8 structure to perform the operation on the data set according to the input data,
 9 wherein executing the at least one SQL statement on the data set performs the
 10 operation;
 11 and
 12 a computer-readable medium to store the obtaining instructions, the structure determining
 13 instructions, and the constructing instructions.

1 71. The computer program product of claim 70 wherein
 2 the computer instructions further comprise:
 3 executing instructions to execute the at least one SQL statement; and
 4 the computer-readable medium further stores the executing instructions.

1 72. The computer program product of claim 70 wherein
 2 the SQL integration object definition comprises a plurality of SQL integration component
 3 definitions;
 4 the data set comprises a plurality of tables;
 5 the SQL integration component definitions comprise:
 6 a parent component definition of a parent/child relationship, wherein
 7 the parent component definition corresponds to a parent table of the tables;
 8 and
 9 the parent component definition comprises a target key definition, the target
 10 key definition corresponding to a target key of the parent table;
 11 a child component definition of the parent/child relationship, wherein
 12 the child component definition corresponds to a child table of the tables; and
 13 the child component definition comprises a foreign key definition, the foreign
 14 key definition corresponding to a foreign key of the child table;

15 the computer instructions further comprise:

16 parent determining instructions to determine that a parent record of the parent table
17 corresponds to a parent of a child record of the child table when the target key
18 of the parent record comprises a value of the foreign key of the child record;

19 and

20 the computer-readable medium further stores the parent determining instructions.

1 73. The computer program product of claim 70 wherein

2 the computer instructions further comprise:

3 instance identifying instructions to identify an instance of the SQL integration object
4 definition in the input data;

5 the SQL integration object definition comprises a SQL integration component definition;

6 the SQL integration component definition corresponds to a table of the data set;

7 the SQL integration component definition comprises a SQL integration field definition;

8 the SQL integration field definition corresponds to a column of the table;

9 the constructing instructions comprise:

10 value determining instructions to use the instance to identify a data value for an input
11 instance of the SQL integration field definition in the input data; and

12 concatenating instructions to concatenate a clause to a first SQL statement of the at
13 least one SQL statement to select a record from the table, the record having
14 the data value for the column;

15 and

16 the computer-readable medium further stores the instance identifying instructions, the value
17 determining instructions, and the concatenating instructions.

1 74. The computer program product of claim 73 wherein

2 the instance identifying instructions further identify a second input instance of the SQL
3 integration object definition in the input data; and

4 the constructing instructions further perform the following:

5 use the second instance to identify a second data value for the SQL integration field
6 definition; and

7 concatenate an OR clause to the first SQL statement to select a second record from
8 the table, the second record having the second data value for the column.

1 75. The computer program product of claim 73 wherein
 2 the computer instructions further comprise:
 3 length determining instructions to determine whether the first SQL statement
 4 comprises a maximum number of SQL clauses, and
 5 when the first SQL statement comprises the maximum number,
 6 construct a new SQL statement;
 7 and
 8 when the first SQL statement does not comprise the maximum number,
 9 perform the concatenating the clause to the first SQL statement;
 10 and
 11 the computer-readable medium further stores the length determining instructions.

1 76. The computer program product of claim 70 wherein
 2 the SQL integration object definition comprises a plurality of SQL integration component
 3 definitions;
 4 the data set comprises a plurality of tables;
 5 the constructing instructions comprise:
 6 selecting instructions to select a current component definition of the SQL integration
 7 component definitions, the current component definition corresponding to a
 8 current table of the tables; and
 9 generating instructions to perform the following:
 10 when the input data comprises an instance of the current component definition,
 11 generate a first SQL statement to select a record from the current table,
 12 the record corresponding to the instance; and
 13 when the input data does not comprise the instance and the input data
 14 comprises a descendant instance of a descendant component definition
 15 of the current component definition, generate a second SQL statement
 16 to select all records from the current table;
 17 and
 18 the computer-readable medium further stores the selecting instructions and the generating
 19 instructions.

1 77. The computer program product of claim 76 wherein

the generating instructions further comprise:

hierarchy determining instructions to determine whether a hierarchy of output data is

limited when the input data does not comprise the instance and the input data

does not comprise the descendant instance; and

child generating instructions to perform the following:

when the hierarchy is not limited,

determine whether the SQL integration object definition comprises a

child component definition of the current component definition,

the child component definition corresponding to a child table of

the tables, and

when the child component definition exists,

generate a third SQL statement to select all records

from the child table;

and

the computer-readable medium further stores the hierarchy determining instructions and the

child generating instructions.

78. The computer program product of claim 70 wherein

the SQL integration object definition comprises a plurality of SQL integration component

definitions;

the data set comprises a plurality of tables;

the SQL integration component definitions comprise:

a parent component definition of a parent/child relationship, the parent component

definition corresponding to a parent table of the data set; and

a child component definition of the parent/child relationship, the child component

definition corresponding to a child table of the data set;

the constructing instructions further comprise:

current component constructing instructions to construct a current component SQL

statement for the child component definition to select a child record of the

child table corresponding to a child instance of the child component definition

from the input data;

parent identifying instructions to use the parent/child relationship for identifying the

parent component definition, the parent component definition determining a

parent result set, the parent result set comprising a parent instance; and

parent selection clause adding instructions to use the parent result set to add a parent selection clause to the current component SQL statement to provide a final SQL statement;

and

the computer-readable medium further stores the current component constructing instructions, the parent identifying instructions, and the parent selection clause adding instructions.

79. The computer program product of claim 78 wherein the parent selection clause adding instructions comprise: parent instance clause generating instructions to generate a parent instance clause to select the child record from the child table, wherein the parent instance clause further selects the child record when the foreign key of the child record comprises a value of the target key from the parent instance;

and

the computer-readable medium further stores the parent instance clause generating instructions.

80. The computer program product of claim 70 wherein the operation comprises upserting the input data into a destination data set; the SQL integration object definition comprises a plurality of SQL integration component definitions; the data set comprises a plurality of tables; a first component definition of the SQL integration component definitions comprises a parent component definition of a parent/child relationship, the parent component definition corresponding to a parent table of the destination data set; a second component definition of the SQL integration component definitions comprises a child component definition of the parent/child relationship, the child component definition corresponding to a child table of the destination data set; the constructing instructions comprise: selecting instructions to select a current component definition of the SQL integration component definitions, the current component definition corresponding to a current table of the tables; and

performing instructions to perform the following for the current component definition:

when the current component definition is the child component definition,

check whether an optimized upsert of a plurality of current component

instances of the current component definition from the input

data can be performed; and

when the optimized upsert can be performed,

construct a first set of at least one SQL statement for

performing the optimized upsert on the child

table, and

when the optimized upsert cannot be performed,

construct a second set of at least one SQL statement to

upsert the current component instances into the

child table;

and

when the current component definition is not the child component definition,

construct a third set of at least one SQL statement to upsert the current

component instances into the current table;

and

the computer-readable medium further stores the selecting instructions and the performing

instructions.

81. The computer program product of claim 80 wherein the performing instructions to construct the first set comprise:
calculating instructions to perform the following:

select a current parent instance of the parent component definition from the input data;

determine a number of child instances of the child component definition from the input data, the child instances corresponding to children of the current parent instance;

retrieve a parent record corresponding to the current parent instance from the parent table;

determine a number of child records from the child table, wherein the child records are children records of the parent record; and

determine a ratio of the number of children instances to the number of child records;

14 and
 15 when the ratio exceeds a predetermined limit,
 16 determine in memory an associated operation of an update operation and an
 17 insert operation for each child instance of the child instances,
 18 and
 19 when the ratio does not exceed the predetermined limit,
 20 access the destination data set for each child instance of the child instances to
 21 determine the associated operation for each child instance of the child
 22 instances;
 23 and
 24 the computer-readable medium further stores the calculating instructions.

1 82. The computer program product of claim 81 wherein
 2 the calculating instructions to determine in memory the associated operation for each child
 3 instance of the child instances comprise:
 4 retrieving instructions to retrieve children records of the parent record from the child
 5 table into memory; and
 6 in-memory instructions to perform the following for each child instance of the child
 7 instances:
 8 select the child instance as a current child instance of the child instances;
 9 use the children records in memory to determine the associated operation for
 10 the current child instance; and
 11 construct an associated SQL statement for performing the associated operation
 12 on the child table;
 13 and
 14 the computer-readable medium further stores the retrieving instructions and the in-memory
 15 instructions.

1 83. The computer program product of claim 70 wherein
 2 the operation is a delete record hierarchy operation;
 3 the SQL integration object definition comprises a plurality of SQL integration component
 4 definitions;
 5 the SQL integration component definitions comprise:
 6 a root component definition, the root component definition corresponding to a root

7 table of a plurality of tables of the destination data set;
 8 a descendant component definition, the descendant component definition
 9 corresponding to a descendant table of the tables, the descendant component
 10 definition corresponding to a descendant of the root component definition;
 11 the constructing instructions further comprise:
 12 root selecting instructions to select a root instance of the root component definition
 13 from the input data;
 14 root marking instructions to mark a root record of the root table corresponding to the
 15 root instance;
 16 descendant selecting instructions to perform the following:
 17 when a cascade delete parameter is set for the descendant component
 18 definition,
 19 select at least one descendant record of the descendant table wherein
 20 the at least one descendant record corresponds to a descendant
 21 of the root record, and
 22 mark the at least one descendant record;
 23 root constructing instructions to construct a root SQL statement to delete the marked
 24 root record from the root table; and
 25 descendant constructing instructions to construct a descendant SQL statement to
 26 delete the at least one marked descendant record from the descendant table;
 27 and
 28 the computer-readable medium further stores the root selecting instructions, the root marking
 29 instructions, the descendant selecting instructions, the root constructing selections,
 30 and the descendant constructing instructions.

1 84. The computer program product of claim 70 wherein
 2 the operation comprises a reverse query operation;
 3 the SQL integration object definition comprises a plurality of SQL integration component
 4 definitions;
 5 the data set comprises a plurality of tables;
 6 the SQL integration component definitions comprise:
 7 a first parent component definition of a first parent/child relationship, the first parent
 8 component definition corresponding to a first parent table of the tables;
 9 a first child component definition of the first parent/child relationship, the first child

component definition corresponding to a first child table of tables;
 and
 the constructing instructions comprise:
 first constructing instructions to construct a first SQL statement for selecting a first
 child record from the first child table, wherein
 the first child record corresponds to an instance of the first child component
 definition from the input data;
 and
 second constructing instructions to construct a second SQL statement for selecting a
 first parent record from the first parent table, wherein the first parent record
 corresponds to a parent of the first child record.

85. The computer program product of claim 84 wherein
 the first parent component definition comprises a target key definition; the target key
 definition corresponding to a target key of the first parent table;
 the first child component definition comprises a foreign key definition, the foreign key
 definition corresponding to a foreign key of the first child table; and
 the first parent record corresponds to the parent of the first child record when the target key
 of the first parent record comprises a value of the foreign key of the first child record.

86. The computer program product of claim 84 wherein
 the SQL integration component definitions further comprise:
 a third component definition, the third component definition corresponding to a third
 table of the tables;
 the first parent component definition corresponds to a second child component definition of a
 second parent/child relationship;
 the third component definition corresponds to a second parent component definition of the
 second parent/child relationship;
 the constructing instructions further comprise:
 third constructing instructions to construct a third SQL statement for selecting a third
 record from the third table, the third record corresponding to a parent of the
 first parent record;
 and
 the computer-readable medium further stores the third constructing instructions.

1 87. A computer program product comprising:
2 computer instructions comprising:
3 obtaining instructions to obtain an operation to be performed on a data set and
4 corresponding input data;
5 constructing instructions to construct at least one SQL statement to perform the
6 operation on the data set according to the input data; and
7 executing instructions to execute each SQL statement of the at least one SQL
8 statement on the data set once said each SQL statement is constructed such
9 that said each SQL statement is executed prior to constructing a subsequent
10 SQL statement of the at least one SQL statement;
11 and
12 a computer-readable medium to store the obtaining instructions, the constructing instructions,
13 and the executing instructions.

1 88. The computer program product of claim 87 wherein
2 the constructing instructions comprise subsequent constructing instructions to construct the
3 subsequent SQL statement using a result set of executing a prior SQL statement.

1 89. A system comprising:
2 obtaining means for obtaining an operation to be performed on a data set and corresponding
3 input data;
4 structure determining means for determining a SQL integration object definition to determine
5 a structure of the data set; and
6 constructing means for constructing at least one SQL statement conforming to the structure to
7 perform the operation on the data set according to the input data, wherein executing
8 the at least one SQL statement on the data set performs the operation.

1 90. The system of claim 89 further comprising:
2 executing means for executing the at least one SQL statement.

1 91. The system of claim 89 wherein
2 the SQL integration object definition comprises a plurality of SQL integration component
3 definitions;
4 the data set comprises a plurality of tables;

5 the SQL integration component definitions comprise:

6 a parent component definition of a parent/child relationship, wherein

7 the parent component definition corresponds to a parent table of the tables;

8 and

9 the parent component definition comprises a target key definition, the target

10 key definition corresponding to a target key of the parent table; and

11 a child component definition of the parent/child relationship, wherein

12 the child component definition corresponds to a child table of the tables; and

13 the child component definition comprises a foreign key definition, the foreign

14 key definition corresponding to a foreign key of the child table;

15 and further comprising:

16 parent determining means for determining that a parent record of the parent table corresponds

17 to a parent of a child record of the child table when the target key of the parent record

18 comprises a value of the foreign key of the child record.

1 92. The system of claim 89 further comprising:

2 instance identifying means for identifying an instance of the SQL integration object definition

3 in the input data;

4 and wherein

5 the SQL integration object definition comprises a SQL integration component

6 definition;

7 the SQL integration component definition corresponds to a table of the data set;

8 the SQL integration component definition comprises a SQL integration field

9 definition;

10 the SQL integration field definition corresponds to a column of the table;

11 and

12 the constructing means comprise:

13 value determining means for using the instance to identify a data value for an

14 input instance of the SQL integration field definition in the input data;

15 and

16 concatenating means for concatenating a clause to a first SQL statement of the

17 at least one SQL statement to select a record from the table, the record

18 having the data value for the column.

1 93. The system of claim 92 wherein
2 the instance identifying means further comprise second instance identifying means for
3 identifying a second input instance of the SQL integration object definition in the
4 input data; and
5 the constructing means further comprise:
6 second value determining means for using the second instance to identify a second
7 data value for the SQL integration field definition; and
8 concatenating means for concatenating an OR clause to the first SQL statement to
9 select a second record from the table, the second record having the second data
10 value for the column.

1 94. The system of claim 92 wherein the further comprising:
2 length determining means for determining whether the first SQL statement comprises a
3 maximum number of SQL clauses, and
4 when the first SQL statement comprises the maximum number,
5 constructing a new SQL statement,
6 and
7 when the first SQL statement does not comprise the maximum number,
8 performing the concatenating the clause to the first SQL statement.

1 95. The system of claim 89 wherein
2 the SQL integration object definition comprises a plurality of SQL integration component
3 definitions;
4 the data set comprises a plurality of tables;
5 the constructing means comprise:
6 selecting means for selecting a current component definition of the SQL integration
7 component definitions, the current component definition corresponding to a
8 current table of the tables; and
9 generating means for performing the following:
10 when the input data comprises an instance of the current component definition,
11 generating a first SQL statement to select a record from the current
12 table, the record corresponding to the instance; and

when the input data does not comprise the instance and the input data comprises a descendant instance of a descendant component definition of the current component definition, generating a second SQL statement to select all records from the current table.

96. The system of claim 95 wherein the generating means further comprise:
 hierarchy determining means for determining whether a hierarchy of output data is limited when the input data does not comprise the instance and the input data does not comprise the descendant instance; and
 child generating means for
 determining, when the hierarchy is not limited, whether the SQL integration object definition comprises a child component definition of the current component definition, the child component definition corresponding to a child table of the tables, and
 generating, when the child component definition exists, a third SQL statement to select all records from the child table.

97. The system of claim 89 wherein
 the SQL integration object definition comprises a plurality of SQL integration component definitions;
 the data set comprises a plurality of tables;
 the SQL integration component definitions comprise:
 a parent component definition of a parent/child relationship, the parent component definition corresponding to a parent table of the data set; and
 a child component definition of the parent/child relationship, the child component definition corresponding to a child table of the data set;
 and
 the constructing means further comprise:
 current component constructing means for constructing a current component SQL statement for the child component definition to select a child record of the child table corresponding to a child instance of the child component definition from the input data;
 parent identifying means for using the parent/child relationship for identifying the

parent component definition, the parent component definition determining a parent result set, the parent result set comprising a parent instance; and parent selection clause adding means for using the parent result set to add a parent selection clause to the current component SQL statement to provide a final SQL statement.

98. The system of claim 97 wherein the parent selection clause adding means comprise:
parent instance clause generating means for generating a parent instance clause to select the child record from the child table, wherein the parent instance clause further selects the child record when the foreign key of the child record comprises a value of the target key from the parent instance.

99. The system of claim 89 wherein the operation comprises upserting the input data into a destination data set; the SQL integration object definition comprises a plurality of SQL integration component definitions;
the data set comprises a plurality of tables;
a first component definition of the SQL integration component definitions comprises a parent component definition of a parent/child relationship, the parent component definition corresponding to a parent table of the destination data set;
a second component definition of the SQL integration component definitions comprises a child component definition of the parent/child relationship, the child component definition corresponding to a child table of the destination data set;
and
the constructing means comprise:

selecting means for selecting a current component definition of the SQL integration component definitions, the current component definition corresponding to a current table of the tables; and
performing means for performing the following for the current component definition:
when the current component definition is the child component definition,
checking whether an optimized upsert of a plurality of current component instances of the current component definition from

the input data can be performed; and
 when the optimized upsert can be performed,
 constructing a first set of at least one SQL statement for
 performing the optimized upsert on the child
 table;
 when the optimized upsert cannot be performed,
 constructing a second set of at least one SQL statement
 to upsert the current component instances into
 the child table;
 when the current component definition is not the child component definition,
 constructing a third set of at least one SQL statement to upsert the
 current component instances into the current table.

100. The system of claim 99 wherein the performing means for constructing the
 first set comprises:
 calculating means for performing the following:
 selecting a current parent instance of the parent component definition from the input
 data;
 determining a number of child instances of the child component definition from the
 input data, the child instances corresponding to children of the current parent
 instance;
 retrieving a parent record corresponding to the current parent instance from the parent
 table;
 determining a number of child records from the child table, wherein the child records
 are children records of the parent record; and
 determining a ratio of the number of children instances to the number of child records,
 and
 when the ratio exceeds a predetermined limit,
 determining in memory an associated operation of an update operation and an
 insert operation for each child instance of the child instances;
 and
 when the ratio does not exceed the predetermined limit,
 accessing the destination data set for each child instance of the child instances
 to determine the associated operation for each child instance of the

child instances.

101. The system of claim 100 wherein
the calculating means for determining in memory the associated operation for each child
instance of the child instances comprise:
retrieving means for retrieving children records of the parent record from the child
table into memory; and
in-memory means for performing the following for each child instance of the child
instances:
selecting the child instance as a current child instance of the child instances;
using the children records in memory to determine the associated operation for
the current child instance; and
constructing an associated SQL statement for performing the associated
operation on the child table.

102. The system of claim 101 wherein
the operation is a delete record hierarchy operation;
the SQL integration object definition comprises a plurality of SQL integration component
definitions;
the SQL integration component definitions comprise:
a root component definition, the root component definition corresponding to a root
table of a plurality of tables of the destination data set;
a descendant component definition, the descendant component definition
corresponding to a descendant table of the tables, the descendant component
definition corresponding to a descendant of the root component definition;
and
the constructing means further comprise:
root selecting means for selecting a root instance of the root component definition
from the input data;
root marking means for marking a root record of the root table corresponding to the
root instance;
descendant selecting means for performing the following:
when a cascade delete parameter is set for the descendant component
definition,

20 selecting at least one descendant record of the descendant table
 21 wherein the at least one descendant record corresponds to a
 22 descendant of the root record, and
 23 marking the at least one descendant record;
 24 root constructing means for constructing a root SQL statement to delete the marked
 25 root record from the root table; and
 26 descendant constructing means for constructing a descendant SQL statement to delete
 27 the at least one marked descendant record from the descendant table.

1 103. The system of claim 89 wherein
 2 the operation comprises a reverse query operation;
 3 the SQL integration object definition comprises a plurality of SQL integration component
 4 definitions;
 5 the data set comprises a plurality of tables;
 6 the SQL integration component definitions comprise:
 7 a first parent component definition of a first parent/child relationship, the first parent
 8 component definition corresponding to a first parent table of the tables; and
 9 a first child component definition of the first parent/child relationship, the first child
 10 component definition corresponding to a first child table of tables;
 11 and
 12 the constructing means comprise:
 13 first constructing means for constructing a first SQL statement for selecting a first
 14 child record from the first child table, wherein
 15 the first child record corresponds to an instance of the first child component
 16 definition from the input data;
 17 and
 18 second constructing means for constructing a second SQL statement for selecting a
 19 first parent record from the first parent table, wherein the first parent record
 20 corresponds to a parent of the first child record.

1 104. The system of claim 103 wherein
 2 the first parent component definition comprises a target key definition, the target key
 3 definition corresponding to a target key of the first parent table;
 4 the first child component definition comprises a foreign key definition, the foreign key

definition corresponding to a foreign key of the first child table; and
the first parent record corresponds to the parent of the first child record when the target key
of the first parent record comprises a value of the foreign key of the first child record.

105. The system of claim 104 wherein
the SQL integration component definitions further comprise:
a third component definition, the third component definition corresponding to a third
table of the tables;
the first parent component definition corresponds to a second child component definition of a
second parent/child relationship;
the third component definition corresponds to a second parent component definition of the
second parent/child relationship; and
the constructing means further comprise:
third constructing means for constructing a third SQL statement for selecting a third
record from the third table, the third record corresponding to a parent of the
first parent record.

106. A system comprising:
obtaining means for obtaining an operation to be performed on a data set and corresponding
input data;
constructing means for constructing at least one SQL statement to perform the operation on
the data set according to the input data; and
executing means for executing each SQL statement of the at least one SQL statement on the
data set once said each SQL statement is constructed such that said each SQL
statement is executed prior to constructing a subsequent SQL statement of the at least
one SQL statement.

107. The system of claim 106 wherein
the constructing means comprise subsequent constructing means for constructing the
subsequent SQL statement using a result set of executing a prior SQL statement.

108. A signal embodied in a carrier wave comprising:
obtaining instructions to obtain an operation to be performed on a data set and corresponding
input data;
structure determining instructions to use a SQL integration object definition to determine a

5 structure of the data set; and
 6 constructing instructions to construct at least one SQL statement conforming to the structure
 7 to perform the operation on the data set according to the input data, wherein executing
 8 the at least one SQL statement on the data set performs the operation.

1 109. The signal of claim 108 wherein the signal further comprises:
 2 executing instructions to execute the at least one SQL statement.

1 110. A signal embodied in a carrier wave comprising:
 2 obtaining instructions to obtain an operation to be performed on a data set and corresponding
 3 input data;
 4 constructing instructions to construct at least one SQL statement to perform the operation on
 5 the data set according to the input data; and
 6 executing instructions to execute each SQL statement of the at least one SQL statement on
 7 the data set once said each SQL statement is constructed such that said each SQL
 8 statement is executed prior to constructing a subsequent SQL statement of the at least
 9 one SQL statement.

1 111. The signal of claim 110 wherein
 2 the constructing instructions comprise subsequent constructing means for constructing the
 3 subsequent SQL statement using a result set of executing a prior SQL statement.

1 112. A signal embodied in a carrier wave comprising data produced by:
 2 obtaining instructions to obtain an operation to be performed on a data set and corresponding
 3 input data;
 4 structure determining instructions to use a SQL integration object definition to determine a
 5 structure of the data set; and
 6 constructing instructions to construct at least one SQL statement conforming to the structure
 7 to perform the operation on the data set according to the input data, wherein executing
 8 the at least one SQL statement on the data set performs the operation.

1 113. The signal of claim 112 wherein the data further comprises second data
 2 produced by:
 3 executing instructions to execute the at least one SQL statement.

1 114. A signal embodied in a carrier wave comprising data produced by:

2 obtaining instructions to obtain an operation to be performed on a data set and corresponding
3 input data;
4 constructing instructions to construct at least one SQL statement to perform the operation on
5 the data set according to the input data; and
6 executing instructions to execute each SQL statement of the at least one SQL statement on
7 the data set once said each SQL statement is constructed such that said each SQL
8 statement is executed prior to constructing a subsequent SQL statement of the at least
9 one SQL statement.

1 115. The signal of claim 114 wherein the data further comprises second data
2 produced by:
3 subsequent constructing means for constructing the subsequent SQL statement using a result
4 set of executing a prior SQL statement.